Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently amended) A computer program product residing on a computer readable storage medium comprising instructions, including a context branch instruction that, when executed, causes a data processing apparatus to:

select another instruction in an instruction stream from one of a branch target instruction associated with a label specified by the context branch instruction and an instruction following the context branch instruction based on a comparison of a current executing ~~context~~ thread number to a ~~context~~ thread number specified by the context branch instruction; and

retrieve the selected other instruction.

2. (Currently amended) The computer program product of claim 1 wherein the branch instruction has one of the following formats:

br=ctx[ctx#, label#], optional_token; and

br!=ctx[ctx#, label#], optional_token;

wherein the label# is a symbolic label corresponding to an address of the other instruction, ~~and~~ wherein ctx# is the ~~context~~ thread number, wherein the syntax "br=ctx" represents a branching operation based on ctx# matching the current ~~context~~ thread number provided by the data processing apparatus, ~~and~~ wherein the syntax "br!=ctx" represents a branching operation based on the ctx# not matching the current ~~context~~ thread number provided by the data processing apparatus, and wherein the syntax "optional_token" includes a value that causes the data processing apparatus to execute a number of instructions corresponding to the value of the optional_token following the context branch instruction before performing a branch operation.

3. (Cancelled)

4. (Currently amended) The computer program product of claim 2 wherein the specified ~~context~~ thread number has valid values of 0, 1, 2, or 3.

5. (Cancelled)

6. (Cancelled)

7. (Currently amended) A method of operating a processor comprising:

performing a comparison of a ~~context~~ thread number of an executing ~~context~~ thread to a ~~context~~ thread number specified by a context branch instruction;

selecting another instruction in an instruction stream from one of a branch target instruction associated with a label specified by the context branch instruction and an instruction following the context branch instruction based on the comparison; and

retrieving the selected other instruction.

8. (Currently amended) The method of claim 7 wherein selecting comprises:

selecting the branch target instruction if the executing ~~context~~ thread number matches the specified context number.

9. (Currently amended) The method of claim 7 wherein the ~~context~~ thread number has valid values of 0, 1, 2, or 3.

10. (Currently amended) A processor that can execute multiple ~~contexts~~ threads and that comprises:

a register stack;

a program counter for each executing ~~context~~ thread;

an arithmetic logic unit coupled to the register stack and a program control store that stores a context branch instruction that causes the processor to:

perform a comparison of a ~~context~~ thread number of an executing ~~context~~ thread to a ~~context~~ thread number specified by the branch instruction;

select another instruction in an instruction stream from one of a branch target instruction associated with a label specified by the context branch instruction and an instruction following the context branch instruction based on the comparison; and

retrieve the selected other instruction.

11. (Currently amended) The processor of claim 10 wherein the context branch instruction causes the processor to select the branch target instruction if the executing ~~context~~ thread number matches the specified ~~context~~ thread number.

12. (Currently amended) The processor of claim 10 wherein the ~~context~~ thread number has valid values of 0, 1, 2, or 3.

13. (Currently amended) A computer program product residing on a computer readable storage medium, for causing a processor that executes multiple ~~contexts~~ threads to perform a function, comprises instructions causing the processor to:

perform a comparison of a ~~context~~ thread number of an executing ~~context~~ thread to a ~~context~~ thread number specified by a branch instruction;

select another instruction in an instruction stream from one of a branch target instruction associated with a label specified by the context branch instruction and an instruction following the context branch instruction based on the comparison; and

retrieve the selected other instruction.

14. (Currently amended) The product of claim 13 wherein the context branch instruction causes the processor to select the branch target instruction if the executing ~~context~~ thread number matches the specified ~~context~~ thread number.

15. (Currently amended) The product of claim 13 wherein the ~~context~~ thread number has valid values of 0, 1, 2, or 3.